

BAB III

LANDASAN TEORI

3.1 Pengertian Rekrutmen

Pengertian rekrutmen menurut Henry Simamora (1997) Rekrutmen (Recruitment) adalah serangkaian aktivitas mencari dan memikat pelamar kerja dengan motivasi, kemampuan, keahlian dan pengetahuan yang diperlukan guna menutupi kekurangan yang diidentifikasi dalam perencanaan kepegawaian.

Proses rekrutmen dapat berasal dari pelamar langsung, lamaran dalam bentuk tertulis, rujukan dari orang lain ataupun iklan, penempatan tenaga kerja serta melalui lembaga/serikat pekerja. Terdapat beberapa cara atau teknik rekrutmen yang umum dilakukan oleh perusahaan atau organisasi. Dilihat dari konsep dasar rekrutmen. Teknik rekrutmen menurut Gomes (2003) digolongkan menjadi dua, yaitu :

1. Teknik Rekrutmen Sentralisasi

Jika rekrutmen disentralisasikan, instansi yang mengelola sumber daya manusia itu bertanggung jawab untuk meminta dari para manager akan perkiraan-perkiraan periodik mengenai jumlah dan tipe pekerja-pekerja baru yang dibutuhkan diwaktu akan datang. Instansi manajemen sumber daya manusia tingkat pusat akan mengeluarkan pengumuman perihal lowongan kerja yang tersedia untuk memenuhi peraturan perundangan *Affirmative Action* yang menghendaki perwakilan proporsional.

2. Teknik Rekrutmen Desentralisasi

Teknik rekrutmen yang didesentralisasikan terjadi di instansi-instansi yang relatif kecil, kebutuhan-kebutuhan rekrutmen terbatas, dan dalam mana setiap instansi memperkejakan berbagai tipe pekerja. Rekrutmen dengan cara ini selalu dipakai untuk posisi khas professional, ilmiah, atau administratif bagi suatu instansi tertentu. Instansi-instansi secara sendiri-sendiri biasanya lebih memilih rekrutmen yang didesentralisasikan karena mereka akan secara langsung mengendalikan proses rekrutmennya

3.2 Pengertian Aplikasi

Definisi aplikasi menurut Sri Widiyanti (2000) Aplikasi adalah sebuah perangkat lunak yang menjadi *front end* dalam sebuah sistem yang digunakan untuk mengolah data menjadi suatu informasi yang berguna untuk orang-orang dan sistem yang bersangkutan.

Aplikasi merupakan alat terapan yang terdapat pada komputer yang difungsikan secara khusus dan terpadu sesuai dengan proses kerja di dalamnya. Ada berbagai jenis dari aplikasi yang digolongkan dalam beberapa kategori , yaitu:

1. *Enterprise Application*

Digunakan untuk organisasi yang cukup besar dengan menghubungkan aliran data dan kebutuhan informasi antar bagian di perusahaan.

2. *Enterprise – Support Application*

Aplikasi yang fungsinya untuk mendukung dari *enterprise*

3. *Individual Worker Application*

Sebagai aplikasi yang biasa digunakan untuk mengolah / mengedit data oleh tiap individu.

4. Aplikasi Simulasi

Aplikasi yang umumnya digunakan untuk melakukan simulasi penelitian dan pengembangan.

3.3 Pengertian Bahasa Pemrograman Visual Basic

Menurut Widodo Budiharto (2003), Visual basic adalah bahasa pemrograman event-driven yang berasal dari Basic. [BUD6]. Event driven artinya program menunggu sampai adanya respon dari pemakai berupa kejadian tertentu, misalnya tombol diklik, atau menu dipilih. Ketika event driven terdeteksi, event yang berhubungan akan melakukan aksi sesuai dengan kode yang diberikan. Basic adalah salah satu bahasa pemrograman yang sudah dikenal oleh pemakai komputer. Bahasa ini dapat dikatakan sebagai bahasa pemrograman dasar atau bahasa pemrograman yang paling mudah yang sesuai dengan namanya. Namun sebenarnya nama basic adalah kependekan dari kata-kata : B (*Beginner's*), A (*All-Purpose*), S(*Symbol*), I (*Intruction*), C (*Code*). Bahasa ini pertama kali muncul pada tahun 1960 dan diperkenalkan oleh Dartmouth College.

Visual Basic merupakan bahasa pemrograman yang menawarkan *Integrated Development Environment* (IDE) visual untuk membuat program perangkat lunak berbasis sistem operasi Microsoft Windows dengan model pemrograman. Bahasa pemrograman Visual Basic merupakan suatu alat untuk mengembangkan program komputer di sistem operasi Windows. Kelebihan dari Visual Basic merupakan bahasa pemrograman yang sudah dukung dengan *Object*

Oriented Programming (OOP), sehingga dalam implementasi pemrograman mudah untuk dimengerti. Kelebihan lain dari Visual Basic adalah bahasa yang user friendly dan mempunyai fasilitas *auto checking* dan *auto correctly* yang dapat memudahkan pengguna untuk mengetahui kesalahan saat implementasi pemrograman.

Visual Basic merupakan bahasa pemrograman yang populer. Karena, selain dari model *Object Oriented Programming* (OOP) yang ada didalamnya tetapi juga karena bahasa pemrograman visual basic tidak lepas dari kemampuannya berinteraksi dengan aplikasi lain yang ada pada sistem operasi Windows dengan komponen ActiveX Control. Dengan adanya komponen ActiveX Control pengguna dapat menggunakan semua model data yang ada pada sistem operasi Windows.

3.4 Pengertian *Document Flowchart*

Menurut Jogiyanto (2005), *flowchart* (bagan alir) adalah bagan (*charts*) yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika dapat didefinisikan sebagai bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem.

Document Flowcharts adalah Bagan alir dokumen atau bisa disebut juga sebagai bagan alir formulir yang merupakan bagan alir yang menunjukkan arus dari laporan dan formulir termasuk tembusan-tembusannya. Dalam pembuatannya, *document flowcharts* memiliki ketentuan-ketentuan yang harus diperhatikan. Salah satunya adalah notasi-notasi yang ada di dalamnya. Notasi yang umumnya ada pada *document flowcharts* antara lain :

Tabel 3.1 Simbol dalam Document Flowcharts

No	Simbol	Nama Simbol	Fungsi
1.		Terminator	Merupakan bentuk dari terminator simbol yang digunakan pada awal pembuatan document flowcharts sebagai mengawali (Start) dan mengakhiri (<i>End</i>) flowcharts.
2.		<i>Manual Process</i> (Proses Manual)	Merupakan notasi dari proses manual yang pada document flowcharts. Dinyatakan sebagai proses manual karena dalam notasi document flowcharts segala bentuk proses masih belum dilakukan oleh komputer.
3.		Dokumen	Merupakan notasi dari dokumen pada <i>document flowcharts</i> . Notasi document ini umumnya digambarkan sebagai bentuk lain dari arsip, laporan atau dokumen lainnya yang berbentuk kertas

No	Simbol	Nama Simbol	Fungsi
4.		<i>Decision</i> (Keputusan)	Merupakan notasi dari suatu keputusan dalam pengerjaan <i>document flowcharts</i> . Dalam penggambaran notasi decision ini selalu menghasilkan dengan keputusan ya atau tidak.

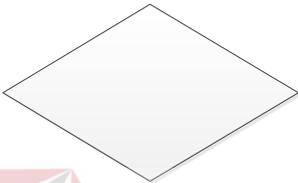
3.5 Pengertian *System Flowchart*

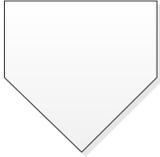
Menurut Jogiyanto (2005), bagan alir program (*system flowchart*) merupakan bagan alir yang mirip dengan bagan alir sistem, yaitu untuk menggambarkan prosedur di dalam sistem. Bagan ini menjelaskan urutan dari prosedur-prosedur yang ada di dalam sistem. Bagan alir sistem menunjukkan apa yang dikerjakan di sistem. Pembuatan *System Flowcharts* memiliki aturan dan ketentuan yang harus diikuti. Seperti halnya dalam pembuatan *Document Flowcharts* sebelumnya, *System Flowcharts* memiliki notasi-notasi sebagai representasi dari proses kerja suatu sistem.

Sebagian notasi dalam *System Flowchart* memiliki kesamaan dengan notasi yang ada pada *document flow* seperti, terminator (*start/end*), dan notasi laporan. Selain kedua notasi tersebut terdapat perbedaan secara bentuk dan fungsinya. Adapun simbol-simbol lain yang ada pada *System Flowchart* akan di jelaskan pada tabel 3.2.

Tabel 3.2 Simbol pada *System Flowchart*

No	Simbol	Nama Simbol	Fungsi
1.		Proses Komputerisasi	Simbol di samping merupakan representasi dari proses sistem. Menggambarkan sistem yang dikerjakan oleh komputer (otomatis).
2.		<i>Database</i>	Gambar disamping adalah representasi dari Database yang mana fungsinya untuk menyimpan data dari proses sebelumnya.
3.		<i>Connector</i>	<i>Connector</i> difungsikan sebagai penunjuk arah aliran dari satu proses ke proses yang lainnya yang saling berkaitan.
4.		<i>Sub-Process</i>	Simbol <i>Sub-Process</i> difungsikan untuk menunjukkan adanya proses yang lebih rinci dari suatu proses utama.

No	Simbol	Nama Simbol	Fungsi
5.		<i>Document</i>	Simbol <i>Document</i> menunjukkan tentang dokumen yang dihasilkan
6.		<i>Decision</i> (Keputusan)	Simbol di samping difungsikan sebagai langkah pengambil keputusan . Keputusan yang ada terkait “ya” atau “tidak” keputusan diambil.
7.		<i>Input/Output</i>	Simbol di samping difungsikan untuk menunjukkan masukan data (input) dan data yang dihasilkan (output).
8.		<i>Connector</i> (<i>On Page Reference</i>)	Untuk menunjukkan hubungan simbol yang saling terkait dalam <i>System Flowchart</i> . Selain itu juga sebagai pengganti garis connector untuk menghubungkan simbol yang saling berjauhan.

No	Simbol	Nama Simbol	Fungsi
9.		<i>Connector (Off- Page Reference)</i>	Fungsinya hampir sama dengan connector (<i>on page reference</i>) perbedaannya adalah untuk menghubungkan simbol yang berada pada halaman yang berbeda.

3.6 Pengertian Data Flow Diagram (DFD)

Data Flow Diagram (DFD) merupakan alat bantu berbentuk grafik yang berfungsi untuk menunjukkan aliran data antar proses dalam sebuah sistem (Kroenke dan Hatch, 1997). Tujuan dari pembuatan DFD adalah untuk mengidentifikasi poin-poin dari proses penting dengan mempresentasikan aliran data antar proses. Empat elemen penting dalam DFD (Kroenke dan Hatch, 1997) adalah external entity, proses, data flow dan data store. Simbol dari elemen DFD dapat dilihat pada Tabel 2.1. Penjelasan dari masing-masing elemen adalah sebagai berikut:

1. External Entity

External entities merupakan sumber atau penerima informasi. *External entities* dapat berupa kantor, departemen, perusahaan, orang maupun agen-agen lain yang berada di luar lingkup sistem yang sedang dimodelkan. Nama dari *external entities* ini diletakkan di dalam simbol *external entities*. Kadang-kadang simbol ini disebut source dan sink. Source adalah sebuah *external entities* yang

menghasilkan sebuah dataflow. Sink adalah sebuah *external entities* yang menyerap atau menyimpan sebuah dataflow.

2. Proses

Proses adalah kegiatan atau kerja yang dilakukan oleh orang, mesin atau komputer dari hasil suatu arus data yang masuk ke dalam proses untuk dihasilkan arus data yang akan keluar dari proses, sehingga sebuah proses harus mengandung minimal satu arus data yang masuk ke dalam proses dan satu arus data yang keluar dari dalam proses.

Proses ini biasanya mempunyai nomor dan nama. Nomor dari proses diikuti dengan huruf E (*Explode*), yang artinya proses masih dapat dipecah lagi menjadi subproses, atau nomor proses diikuti dengan huruf P (*Primitive*), yang artinya proses sudah tidak dapat dipecah menjadi subproses lagi. Sedangkan nama dari proses terdiri dari kata kerja dan diikuti oleh kata benda yang ditulis dalam simbol proses tersebut.

3. Dataflow (Arus Data)

Dataflow menggambarkan perpindahan data antar proses, external entities, dan data store dalam suatu DFD. Dataflow ini berisi data, dengan nama atau isi dari setiap perputaran data yang ditulis pada simbol tanda panah. Dataflow merupakan data tunggal atau merupakan formulir-formulir, laporan-laporan, maupun dokumen-dokumen yang lain. Isi dari suatu dataflow biasanya didokumentasikan secara terpisah dari DFD dan kamus data.

4. Data store

Data stores adalah tempat untuk menyimpan data baik secara manual maupun secara otomatis. Dalam simbol data store terdapat penomeran dan nama. Untuk penomeran ditulis pada bagian kiri data *store*, D adalah simbol data/basis data kemudian diikuti dengan nomor data. Sedangkan penulisan nama pada data stores ditulis dalam simbol data *stores*. Tempat penyimpanan dapat berupa filing cabinet, lemari, file komputer, atau data tersebut dapat terletak didalam pikiran seseorang.

Tabel 3.3. Notasi Data Flow Diagram

No.	Nama notasi	Gambar notasi
1	<i>External entity</i>	
2	Proses	
3	<i>Data flow</i> (Arus Data)	
4	<i>Data store</i>	

3.7 Pengertian Manajemen Basis Data (DBMS)

Menurut Connolly (2005), Sistem Manajemen Basis data (DBMS) merupakan suatu sistem perangkat lunak (*software*) yang membantu pemakai dalam mendefinisikan, menciptakan, mengatur dan mengontrol akses pada suatu basis data.

DBMS menyediakan beberapa fasilitas sebagai berikut :

1. *Data Definition Language* (DDL)

Memungkinkan user untuk membuat spesifikasi tipe data, mendefinisikan basis data, struktur data dan data constant.

2. *Data Manipulation Language* (DML)

Memungkinkan pemakai untuk memasukkan, memperbaharui, menghapus, dan mengirimkan atau mengambil data dari basis data.

Ada lima komponen Sistem Manajemen Basis data (DBMS), yaitu :

1. Hardware (Perangkat Keras)

Perangkat keras yang dibutuhkan untuk menjalankan DBMS dan aplikasi-aplikasi. Contoh: *single personal computer*, *single mainframe*, atau komputer yang menggunakan jaringan.

2. Software (Perangkat Lunak)

Komponen perangkat lunak terdiri dari perangkat lunak DBMS itu sendiri dan program-program aplikasi, bersama dengan sistem operasi, termasuk perangkat lunak jaringan. Contoh: VB.Net

3. Data

Data merupakan komponen yang paling penting dari DBMS, khususnya dari sudut pandang pemakai akhir mengenai data.

4. Prosedur

Cara untuk menjalankan sistem, seperti bagaimana masuk ke dalam DBMS memulai dan menghentikan DBMS, bagaimana membuat data *backup* dari basis data.

5. Manusia

Komponen terakhir adalah manusia yang terlibat dengan sistem, termasuk didalamnya adalah *Database Administrator* (DBA), perancang basis data, pengembang aplikasi, dan pemakai akhir.

